

## **REMARKS**

In the Official Action mailed on **25 January 2008**, the Examiner reviewed claims 1-30. Examiner rejected claims 1-8, 11-18 and 21-28 under 35 U.S.C. § 103(a) based on Padula (U.S. Patent No. 6,330,486, hereinafter “Padula”), and Jones (U.S. Patent No 6,832,192, hereinafter “Jones”). Examiner rejected claims 9-10, 19-20 and 29-30 under 35 U.S.C. § 103(a) based on Padula, in view of Gibson (U.S. Patent No. 5,812,688, hereinafter “Gibson”).

### **Rejections under 35 U.S.C. § 103(a)**

Examiner rejected independent claims 1, 11, and 21 under 35 U.S.C. § 103(a) as being unpatentable in view of Padula in combination with Jones. More specifically, Examiner averred that Padula in combination with Jones disclosed intercepting calls to a non-three-dimensional API to generate audio from a non-three-dimensional application. Applicant respectfully because neither Padula nor Jones disclose intercepting parameters associated with audio use from an application, wherein the application does not include support for three-dimensional sound. Moreover, nothing in Padula in any way suggests intercepting an API call to a non-three dimensional API from a non-three-dimensional aware application so that sound can be produced from the non-three-dimensionally aware application in the 3D environment. The following sections separately detail Applicant’s arguments for each of these points.

### **Non-Three-Dimensionally Aware Application**

Examiner avers that Padula column 2, lines 20-24 discloses the limitations “wherein the application does not include support for three-dimensional sound” (see page 2 of the Office Action letter mailed on 25 January 2008). However, *the VRML-enabled applications* that Padula discloses already support three-

dimensional sound. Specifically, Padula column 1, line 66 to column 2, line 4 discloses:

“Properly rendering a VRML world requires a **special web-browser or application** such as Silicon Graphics WebSpace or CosmoPlayer. These applications interpret the VRML scene graph and render a complex three-dimensional world providing the user with a virtual camera looking into the virtual world.

The virtual cameras of **the rendering engines** are not limited to a single fixed length lens...”

Thus, rendering VRML requires an application (with a rendering engine) which already supports three-dimensional rendering. Padula column 2, lines 18-24 then states:

“**These rendering engines do not just render a three-dimensional visual environment, they also render a three-dimensional aural environment.** The single-channel (i.e., monophonic) sounds generated by the sound emitter nodes are rendered stereophonically to simulate three-dimensional positioning of the audio source.”

Thus, Padula discloses using the VRML rendering engine to render three-dimensional audio. In other words, Padula discloses using the same VRML rendering engine to render both three-dimensional video and three-dimensional audio. Hence, Padula does not disclose an application which does not include support for three-dimension sound because Padula discloses an application that includes support for both three-dimensional video and three-dimensional sound (i.e., the VRML rendering engine).

Note that the fact that *the sound source* of Padula (i.e., the single-channel sounds generated by the sound emitter nodes) is not three-dimensional **does not establish** that Padula discloses *applications* that do not include support three-dimensional sound for at least the reason that a *sound source* is not the same as an

*application* which uses the sound sources to produce sounds. Applicant respectfully points out that the claim limitations at issue are directed to the *applications* (e.g., the legacy application which do not support three-dimensional sound) and **not** to the sound sources.

Thus, there is nothing in Padula that suggests that the applications (i.e., the VRML rendering engine) do not support three-dimensional sound.

In contrast, embodiments of the present invention enable three-dimensional sound in **legacy applications which do not already support three-dimensional sound**.

#### **Intercepting API Calls from Non-Three-Dimensionally Aware Applications**

Examiner also avers that Padula discloses intercepting information about audio use by intercepting calls from a non-three dimensionally aware application to a non-three-dimensional audio API. Applicant respectfully disagrees. As described above, Padula is limited to a three-dimensional rendering engine that takes into account a three-dimensional sound object's relative position when generating sound. Nothing in Padula in any way suggests intercepting an API call to a non-three dimensional API from a non-three-dimensionally aware application so that sound can be produced from the non-three-dimensionally aware application in a three-dimensional environment.

More specifically, as shown in FIG. 1 and described in col. 5, lines 27-38 of Padula:

“[A] rendering engine for a nodal modeling language (e.g., VRML) is modified to provide a correct acoustic perspective in accordance with the present invention. Referring to FIG. 1, the **audio renderer element of the rendering engine is executed at regular intervals** (30 times a second in CosmoPlayer) as shown at 10. A **scene manager provides the audio renderer with the location coordinates** and other characteristics of the

relevant node and, as shown at decision block 11, only if the location has transformed (i.e., the observer's viewpoint or the node moved) is the acoustic perspective changed as shown at 12” (emphasis added).

As clearly described in this section of Padula, the node itself is not *calling* the “audio renderer element of the rendering engine.” Instead, the audio renderer is executed periodically (e.g., 30 times a second) to process a set of audio nodes from the scene graph. Padula is expressly limited to “altering characteristics of nodes which are **already used by the audio renderer** (e.g., intensity and location)” for subsequent processing by the audio renderer (see Padula, col. 5, lines 40-53, emphasis added).

Examiner cited Padula, col. 3, lines 60-67 in rejecting original claim 6, (which contained a claim related to the interception of the API call from the non-three-dimensionally aware application) (see Office Action, page 4). However, Applicant respectfully points out that the cited section of Padula generally describes only “a third element” (which can be separate from or part of the rendering engine) for *computing the location* of the audio in a 3D environment:

“The present invention provides a third element interposed between the computer-readable representation and the rendering engine (although it may be integrated directly into the rendering engine if desired). In the present invention, the **location coordinates of audio signal sources are correlated with the graphical representation before audio signal rendering**. In particular, the difference between the virtual camera field of view and the user's field of view is determined. This difference is then used to warp the locations of audio sources in proportion to this determined difference, and these warped locations are provided to the rendering engine to render the audio portion of the virtual environment (emphasis added).”

Nothing in the cited section or anywhere in Padula in any way suggests intercepting an API call to a non-three dimensional API from a non-three-

dimensional aware application so that sound can be produced from the non-three-dimensionally aware application in the three-dimensional environment.

In contrast, in embodiments of the present invention, the 3D graphics system and 3D sound system enable the positioning of sound from non-three-dimensionally aware applications (i.e., legacy 2D applications) within the 3D environment (thereby enabling a user to receive spatial audio cues from these legacy 2D applications) (see instant application, par. [0010]-[0011]). In these embodiments, the application execution environment is reconfigured so that calls to a non-three-dimensionally aware audio API (i.e., a legacy audio API) from a non-three-dimensionally aware application **are intercepted using wrapper code and processed using the 3D window manager and 3D audio code** (i.e., the three-dimensional sound system) (see instant application, par [0036]-[0038]). In these embodiments, the 3D window manager and the non-three-dimensionally aware application execute in different processes (i.e., the non-three-dimensionally aware application is *a separate process* from any 3D rendering process) (see instant application, par. [0037]).

More specifically, in embodiments of the present invention, when the non-three-dimensionally aware application attempts to create sound by making a call to a non-three-dimensional audio API, the wrapper code intercepts the call to the non-three-dimensional audio API and forwards the call to the 3D audio code. The 3D window manager calculates an apparent source location for audio produced by the non-three-dimensionally aware application based upon the location of the display object representing the non-three-dimensionally aware application in the 3D environment. These embodiments then position the audio from the non-three-dimensionally aware application in the 3D environment using the 3D audio code based on the calculated apparent source location.

Nothing in Padula in any way suggests intercepting an API call to a non-three dimensional API from a non-three-dimensional aware application so that

sound can be produced from the non-three-dimensionally aware application in the 3D environment.

Applicant has amended claims 1, 11, and 21 to clarify that embodiments of the present invention intercept a call to generate audio to a non-three-dimensional audio API from a non-three-dimensionally aware application and use the call to the non-three-dimensional audio API and the calculated audio source location in a three-dimensional sound system to position audio from the application in a three-dimensional sound space. These amendments find support in pars. [0010] and [0036]-[0038] of the instant application. Applicant has also cancelled claims 6-8, 16-18, and 26-28 without prejudice. No new matter has been added.

Hence, Applicant respectfully submits that independent claims 1, 11, and 21 as presently amended are in condition for allowance. Applicant also submits that the dependent claims that depend from these independent claims, are for the same reasons in condition for allowance and for reasons of the unique combinations recited in such claims.

## **CONCLUSION**

It is submitted that the application is presently in form for allowance.  
Such action is respectfully requested.

Respectfully submitted,

By /Anthony Jones/  
Anthony Jones  
Registration No. 59,521

Date: 23 April 2008

Anthony Jones  
Park, Vaughan & Fleming LLP  
2820 Fifth Street  
Davis, CA 95618-7759  
Tel: (530) 759-1666  
Fax: (530) 759-1665  
Email: tony@parklegal.com